RT-LAB
Quickstart Guide

www.OPAL-RT.com

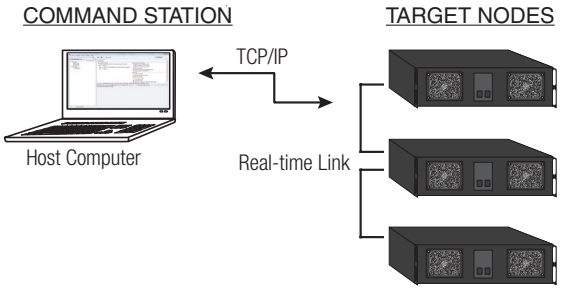
CONTENTS

ABOUT RT-LAB

RT-LAB is a real-time simulation platform for high-fidelity plant simulation, control system prototyping, and embedded data acquisition and control. Its unique distributed processing capability allows you to quickly convert your Simulink™ or Systembuild™ models to high-speed, real-time simulations, over one or more target PC processors.

HOW RT-LAB WORKS

RT-LAB easily integrates Simulink or SystemBuild models in real-time simulations over one or more PC processors, particularly for Hardware-in-the-Loop (HIL) applications. It runs on a networked PC configuration consisting of a Command (or Host) Station, Target Node(s), Communication Links (real-time and Ethernet), and I/O boards.



COMMAND STATION: A Windows or RedHat PC, on which the RT-LAB software is installed, that serves as the development system and user interface. It allows users to prepare the model for distributed real-time execution, control the compilation and execution, and interact with the simulation at run-time.

TARGET NODE: A PC or cluster of PCs where the simulation runs. For real-time simulation, this requires a Real-Time Operating System (RTOS) such as QNX or RedHat Real-Time Linux. The Command Station communicates with Target Node(s) using Ethernet, and the Target Node(s) communicate with each other using real-time data links such as IEEE 1394 (Firewire at 800 Mbits/s or PCI Express Dolphin Communication fabric at 20 Gbits/s). For Hardware-in-the-Loop simulations, Target Nodes connect to the real world through I/O boards (analog and/or digital) and record data at high speed on their local hard drives, if required.

Each target node can contain one or two processor chips that are compatible with Intel or x86 standards and with shared-memory communications. Each processor can then contain a one, two, four or six processor core.

Target System: Includes several target computers

Target Computer: Includes several processor nodes

INTRODUCTION

Now that you have successfully installed your RT-LAB software following the instructions provided in the RT-LAB Installation Guide (C:\OPAL-RT\RT-LAB10.x.x\help\pdf\ RT-LAB_IG), you will need to be up and running quickly. This Quickstart Guide is intended to guide you through the process of setting up your simulation, using *rtdemo1* as the model; it is not intended to be used for detailed training or advanced use. If you require more detailed or advanced procedures, consult either the User Manual or the Cheat Sheets provided in the Help Files.

Software updates are available from the OPAL-RT website at http://www.OPAL-RT.com/downloads. Clicking on the Software name and version will display the MATLAB (and other applicable third party software) versions supported by the latest OPAL-RT software update.

If you encounter any difficulties not resolved through this document, please contact OPAL-RT Technical Support:

The OPAL-RT Team
www.OPAL-RT.com/support/support-request
Tel: +1-514-935-2323 (8:30am to 5:30pm EST/DST)
Toll free: 1-877-935-2323

SOFTWARE REQUIREMENTS

- MATLAB R2007b and higher
- Windows XP and higher

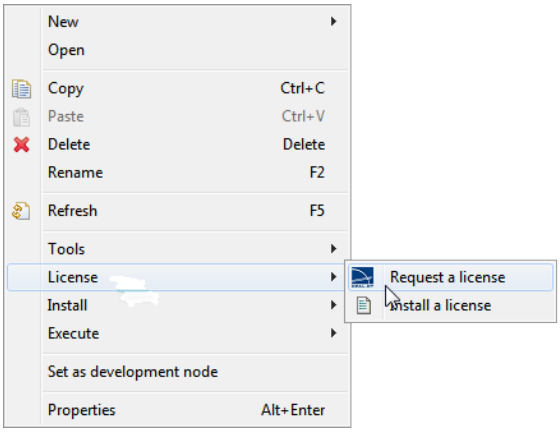
HARDWARE REQUIREMENTS

- OPAL-RT simulator
- Dedicated processor target (PC)

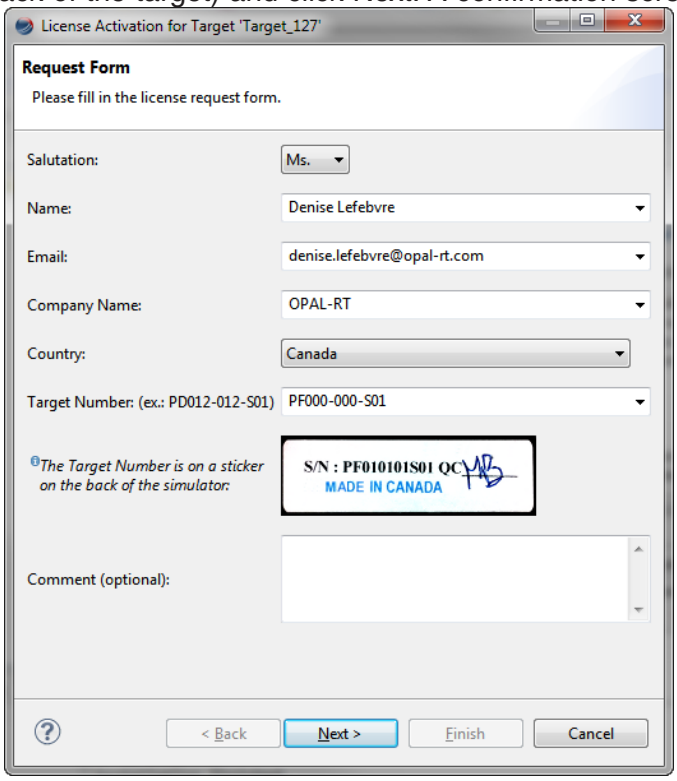
GETTING STARTED

RT-LAB licenses are now issued for targets, rather than command stations. Therefore, before you begin using RT-LAB, you must load any and all targets and obtain your software license. Follow these simple instructions:

1. Click to select the desired target, then right-click and select “Request a License” (this opens a license request form)



2. Fill in the license request form (the target number should automatically appear; if it does not, refer to the sticker on the back of the target) and click Next. A confirmation screen appears.



3. If the information is correct, click Finish to submit the license request. You will receive your software license via email.
4. When you receive the license file, save it in the OPAL-RT directory (ex.: C:\OPAL-RT\LicServer), then open RT-LAB, select the target, right-click it and select ‘Install License’.

OPEN RT-LAB

Once you have installed RT-LAB, there is an icon displayed in the taskbar (lower right-hand corner of the Windows screen). Simply double click the RT-Lab icon (shown) to open RT-LAB.

THE WELCOME SCREEN

The first time you open RT-LAB, the first screen is the welcome screen The screen displays 3 icons: Overview, Tutorials and What's New.

Icon	Description
	Overview provides access to the RT-LAB user manual
	Tutorials provides access to 3 different tutorial subjects: "Create a new project using a template model", "Add an existing model to a project" and "Six steps to Real-Time"
	What's New provides links to the OPAL-RT website and to various trade articles and press releases.
	The Workbench icon (top right hand of the screen) closes the Welcome page and displays the RT-LAB main page. Clicking on this minimizes the Welcome toolbar to the lower right-hand portion of the RT-LAB screen.

MAIN RT-LAB WINDOW

The main RT-LAB window allows you to manage all aspects of your simulation from within a simple directory tree structure.

Remember that you can click on the Help menu at any time to open the "Cheat Sheets" (task oriented instructions) option, that can guide you through this process.

Tree Structure Icons

Icon	Description
	Indicates that the target (circled platform letter) is set as a development node
Q,R,W	Indicates the target's platform (Q, Qnx; R, Redhat, W, Windows)
	Indicates that the target is not responding, not running.

Help files and "Cheat Sheets" are available by clicking on the Help menu. Installation and user guides can be found in the OPAL-RT directory created when the software was installed (C:\ OPAL-RT\RT-LAB10.x.x\help\pdf)

USING THE TOOLBAR

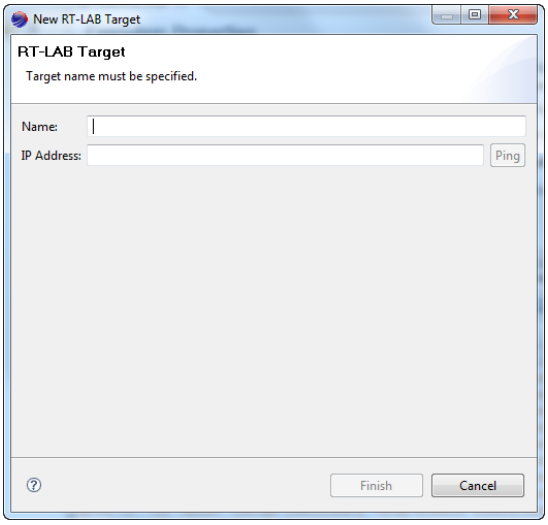
The main RT-LAB screen provides a convenient toolbar that lets you perform functions with a simple click. Here is a brief description of the available buttons:

Icon	Name	Description
	New	Provides access to New Project, New Target, New Model and Other windows
	Save	Saves any changes made to the project, model or target.
	Build	Builds the model
	Assign	Assigns the model to a target
	Load	Loads the model and prepares it for execution
	Execute	Executes the model
	Pause	Pauses the model execution
	Reset	Resets the model (stops any current execution)
	Take Snapshot	Captures model states at current point and allows future simulations to start from that point
	Matlab (Rxx)	Opens Matlab (if more than one version is installed, they will be listed in a drop-down menu to open a specific version)
	Run Python script	Records actions to be used as an automatic script written in Python (ex.: repetitive actions can be recorded and performed automatically by the script). See the Help Files and “Cheat Sheets” for details.
	Parameters Control	Opens the Parameters Control window (see page 13) to configure model parameters
	Probe Control	Opens the Probe Control window (see page 14) to configure data acquisition parameters
	Monitoring Viewer	Opens the monitoring viewer to display processor activities
	Scopeview	Opens Scopeview to display and analyze waveforms

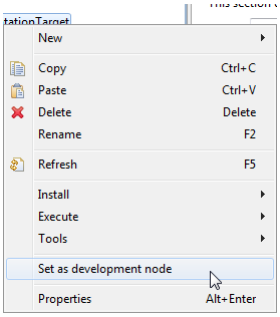
BASIC SETUP

Create a New Target Node

The first step is to create a new target node:

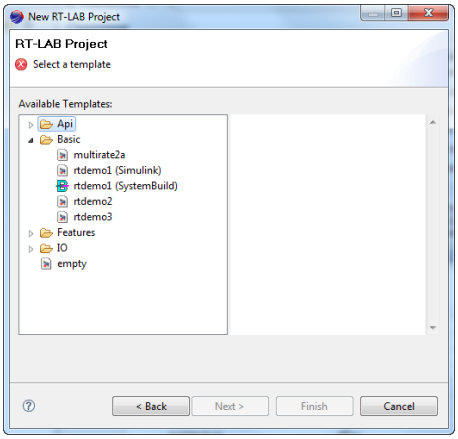


- Click on the New button and select New Target.
- Enter a name and the specific IP address for the target (all targets are shipped with a label affixed to the housing with the default IP address. See your network administrator to obtain a different IP address.)
- Click on Ping to ensure that the target is available and click “Finish” when ping is complete. The New Target window closes and the target now appears in the left-hand directory structure
- Select the development node: right click on the target name and select the “Set as development node[1]” option. The model will be built on this target node.
- It is always a good idea to ensure that the target node runs the correct version of RT-LAB: simply right click on the target name, select Install and then select RT-LAB



Create a New Project with Model

Now, create a new project[2] (in this example, using sample model “rtdemo1”):



- Click on the New button and select New Project to open the new project wizard.
- Enter a project name (ex.:DocTest) then click **Next**.
- Browse the template directory to select the model to add to the project: open the Basic folder and select rtdemo1 (Simulink).
- Click **Finish** to create the new RT-LAB project in the project explorer.

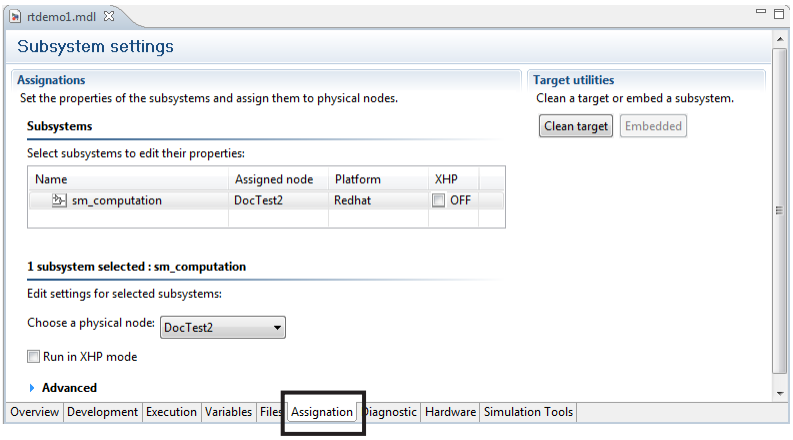
Build the model.

- Select the model in the project explorer. Click the build button on the toolbar to start the compilation.

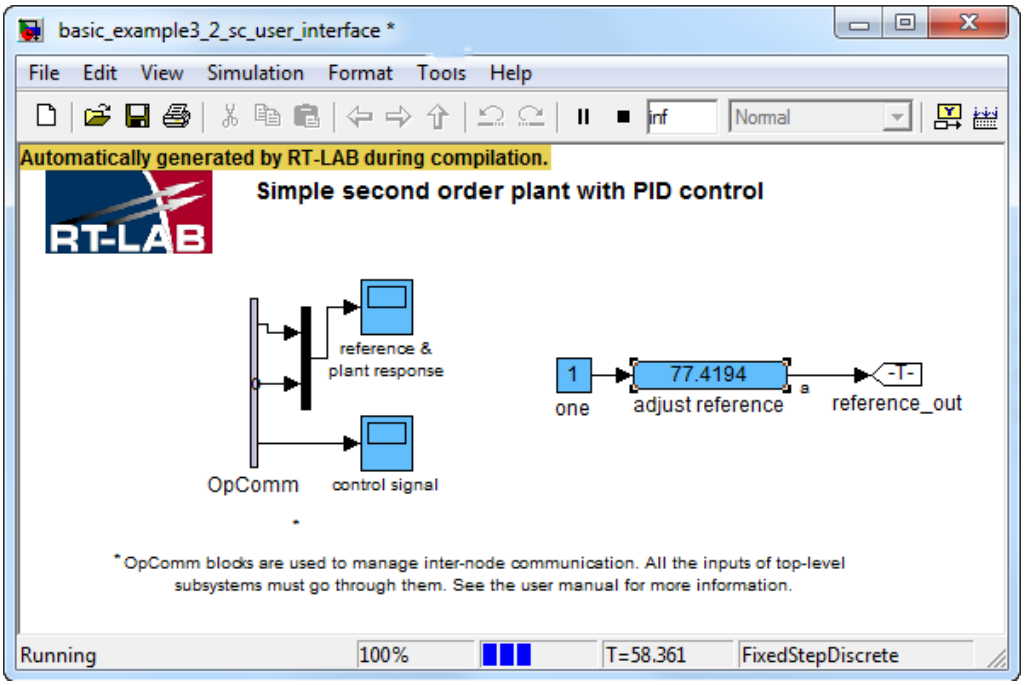
1: The development node is the target node on which the model will be built and executed.
2: A project is simply the framework that will contain all the models required for the simulation.

Assign Subsystems to targets

- Open the Assignation tab
- Assign a target node to the model subsystem(s), if a target other than the default development node is desired.



Load and Execute the model



This transfers and runs the model on the specified target. Select the model in the project explorer and click on the load button on the toolbar. The simulation is transferring to the target, which will be in Pause mode:

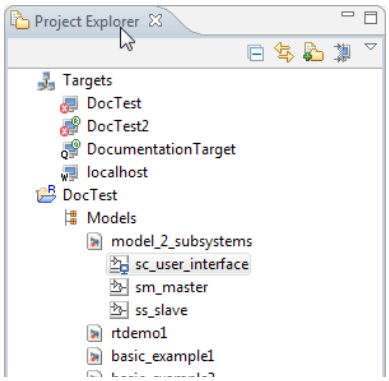
- Execute the model by clicking on the execute button. The simulation is now running on the target.

NOTE: the highlighted yellow text **Automatically generated by RT-LAB during compilation.** indicates that this model was generated by RT-LAB, which means that this is a running model. When the model is generated (played) in Simulink, this notice does not appear, which means that the model is running offline.

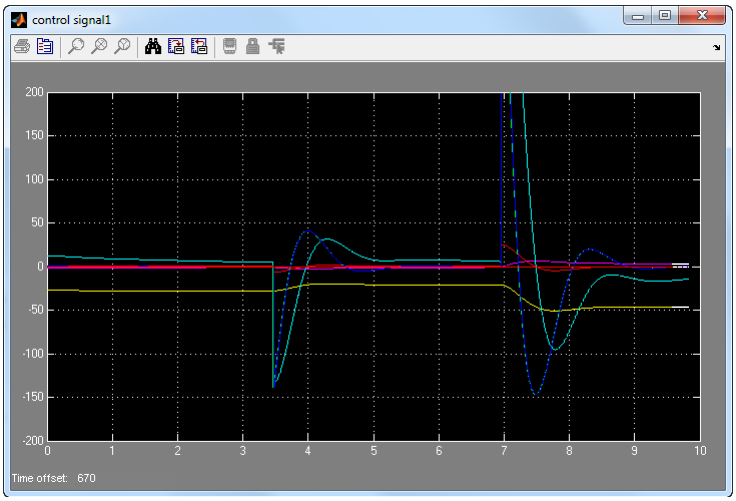
Open and execute the console

Open the generated console to display the simulation results:

- In the project explorer, expand the tree below the model name.
- Double click on the console name (subsystem name beginning with SC). MATLAB opens and displays the generated console.
- Then click on the play button (inside Simulink) to start the console. The console is now receiving and sending data to the simulation.



To view the simulation results on an oscilloscope, simply double click the desired scope block in Simulink



Reset the Model

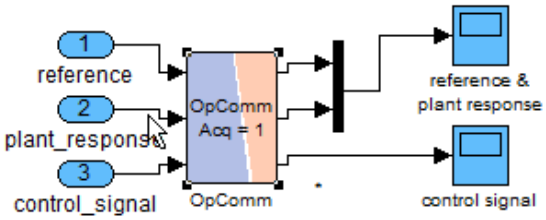
Stop the simulation and reset the model running on the target.

- Click on the reset button on the toolbar to stop the simulation on the target.

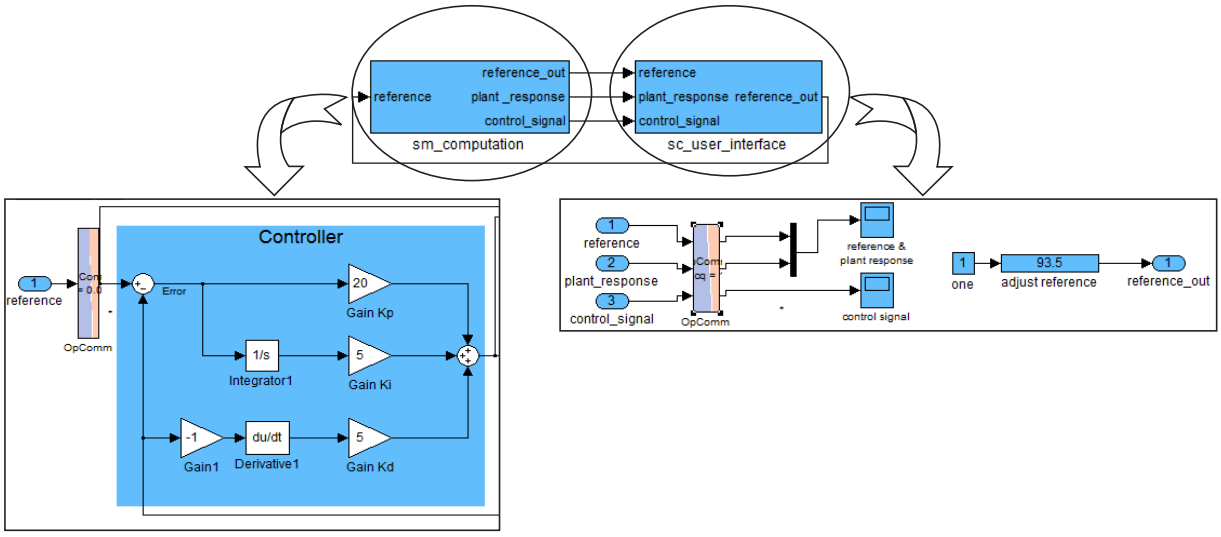
UNDERSTANDING RTDEMO1 CHARACTERISTICS

OpComm Block

The rtdemo1 model used throughout this guide contains basic Simulink blocks and a critical custom block - OpComm. OPAL-RT has designed the OpComm block to enable communication between the Command Station and Target(s) in a distributed simulation.



Each subsystem must contain its own OpComm block, as shown in the image below.



Subsystems

Models contain top level subsystems; master (SM), slave (SS) and console (SC) subsystems for computations. All inputs to top-level subsystems must first go through an OpComm block before they can be used by RT-LAB.

In the computation subsystems (SM or SS):

- One OpComm receives real-time-synchronized from other computation subsystems
- One OpComm receives signals asynchronously from the console subsystem

In the console subsystem (SC subsystem):

- One or more OpComm blocks may be inserted to receive signals from other target nodes (in the case of multi-node systems)
- Acquisition groups[3] can be defined for each OpComm blocks. .

3: RT-LAB enables you to set different acquisition groups that group can include many parameters, including the number of signals per frame to receive, decimation factor, etc. These parameters can be defined separately for each signal using the Probe Control Panel.

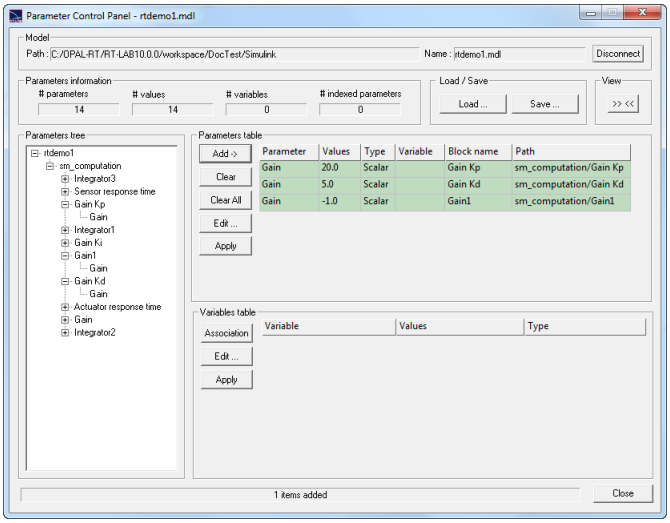
USING SIGNALS AND PARAMETERS

Signals, in Simulink, are the values exchanged between the various blocks. RT-LAB's SC subsystem lets you send and receive signals from your model by wiring those signals to the SC subsystem, which can then be displayed.

Parameters are values that are internal to a block. These values are accessible from the block's mask, displayed by double-clicking on the block in Simulink. RT-LAB allows you to modify certain parameters ***while the model is running***.

Use the RT-LAB main Parameters Control and Probe Control panels to adjust model parameters and tune your view of the running simulation.

Parameters Control



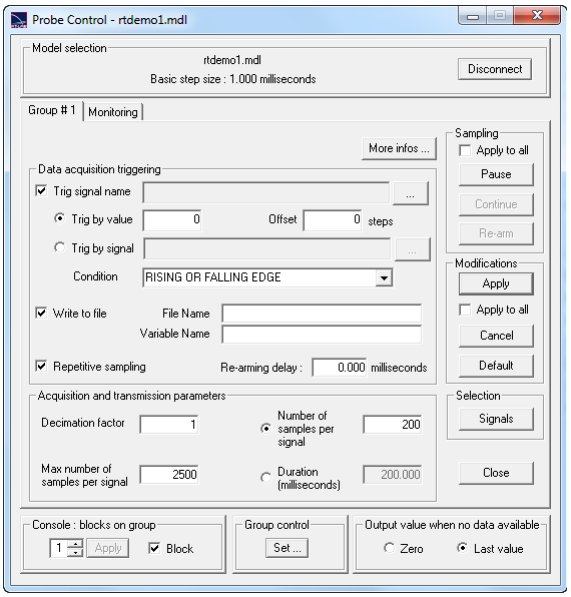
The Parameters Control panel allows you to select model parameters and change their values. You can also use variables to define parameters, allowing you to use the same parameter in many parts of the model and change its value in only one place.

- Select the desired parameters in the tree on the left
- Click on **Add** to add the parameter to the table. The parameters can now be edited in the "Parameters Table", in the right-hand portion of the window.

Probe Control Panel

The Probe Control Panel lets you control acquisition and transmission of signals from the real-time simulation to the user interface. Changes to Probe Control settings will not affect model execution.

Data acquisition triggering



Trig signal name: sets the name of the signal to be triggered.

Trig by value: specifies the value that must be reached by the trigger signal to trigger acquisition.

Trig by signal: specifies a triggering signal instead of a constant value

Condition: condition to be satisfied so that the acquisition system is triggered. The different types of condition are:

- Rising or falling edge
- Rising edge (the threshold is crossed on a positive slope, that is from below)
- Falling edge (the threshold is crossed on a negative slope, that is from above)
- Trigger signal >= trigger level
- Trigger signal <= trigger level.

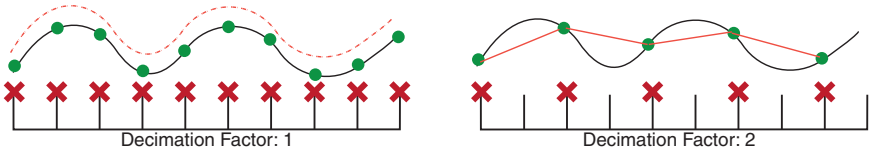
Write to file: saves acquisition group signals on the target node's hard drive rather than sending them to the Windows NT Console on the host computer. Signals are saved in MATFILE version 4, using the name supplied in the FileName field.

Repetetive sampling: enables you to choose repetitive (automated) sampling or non-repetitive (manual control) sampling. If you choose non-repetitive, the Re-arm button enables you to manually re-arm data acquisition.

Acquisition and transmission parameters

Dynamically sets a trigger on the acquisition group (that are applied to OpComm blocks in the sc_ subsystem). Data will be acquired only when the specified signal is triggered. The trigger is defined using the following fields. Click on Signals to open the Dynamic Signal Selection window and configure the Acquisition Group. The following settings determine how much data will be gathered before it is sent to the command station.

Decimation factor:	determines the data acquisition frequency on the computation node
Max number of samples per signal:	determines the size of the acquisition window per signal (number of data points gathered in buffer before sending
Duration:	determines the acquisition duration, in milliseconds



With a decimation factor of 1 (or a timestep of 1 ms), you are able to capture all simulation data, providing a better overview. With a decimation factor of 2, there is a risk of missing parts or all of a range of data, as shown above.

These settings may require repeated adjustments to obtain optimal results, particularly when using synchronized execution modes:

The total number of data points is equal to the number of samples per signal x the number of signals

CONTACT

OPAL-RT Corporate Headquarters

1751 Richardson, Suite 2525
Montréal, Québec, Canada
H3K 1G6
Tel.: 514-935-2323
Toll free: 1-877-935-2323

Technical Services
www.OPAL-RT.com/support

Note:

While every effort has been made to ensure accuracy in this publication, no responsibility can be accepted for errors or omissions. Data may change, as well as legislation, and you are strongly advised to obtain copies of the most recently issued regulations, standards, and guidelines. This publication is not intended to form the basis of a contract.

QR13-10066-RT1
05/2013
© OPAL-RT Technologies Inc.

MATLAB, Simulink and Real-Time Workshop are trademarks of The Mathworks, Inc. LabVIEW, MATRIXx, SystemBuild and AutoCode are trademarks of National Instruments, Inc. QNX is a trademark of QNX Software Systems Ltd. FireWire is a trademark of Apple Corporation, Inc.

RedHawk is a trademark of Concurrent Corp. All other brand and product names are trademarks, registered trademarks or service marks of their respective holders. All rights reserved.